SAN DIEGO Startup week

@SDStartupWeek



SAN DIEGO Startup week

Choosing the right CI/CD system for your team

Fadee Kannah CTO at CourseKey

Thank You 2019 Sponsors!







XP Development Process Prototype

Scrum

Kanban

Software Development Life Cycle (SDLC)



The evolution of the development process



The evolution of the development process



Definitions

- Continuous Integration (CI) is the **process** of integrating the development team's work into a shared repository as soon **any change** is made.
- Continuous Delivery (CD) is the ability to get changes into **production** safely and quickly in a <u>sustainable</u> way.



The complete process



Why is a development process important?



CommitStrip.com

The benefits of the process

- Code management
- Defects detection
- Visibility to data and metrics
- Team process!

How can this process be implemented

- Manually
- Automated
- Buy tools that specialize in this process
- Design your own system

Manual

Pros:

- Ready out of the box
- Simple to follow

Cons:

- Increases in complexity
- Error prone
- Time consuming



Automated

Pros:

- Simplifies the manual process
- Allows anyone to use it

Cons:

- Time/ Resources investment
- Visibility



Buying

- What to buy
- Where to buy
- How much it will cost
- What it takes to maintain

Buying

List of Continuous Integration services

build passing

Subscribe to receive notificatons.



Telegram: t.me/sqaunderhood

There are a lot of cloud continuous integration services. All of them have different set of functionality, some of them require payment, some of them are free. I have created a list of such services to make easy comparison of them and choose more suitable for you.

Some of CI services has open source code thus you can setup them in standalor mode. Pay attention to the column "Features".

There is a similar comparison on Wikipedia.

Name	Description	Features	Supported repositories	Documentation	Price
	Self-Hosted,	Languages:			

Buying

Default Branch

✓ master	# 424	4 passed out an hour ago	-⊙- 4df3ee2	~	\checkmark	~	\checkmark	~
Active Branches	1	sudo: fal	lse					
★ greenkeeper/ws-7.0.0	2	language:	node_js	×				
✓ greenkeeper/ws-6.2.1	3	node_js:		~				
Inactive Branches	4	- "6"						
✓ greenkeeper/eslint-config-google-0.13.0	5	- "8"		~				
	6	- "10"						

Designing

- Where do you start
- How do you do this



Designing

Software Engineering Blogs - ------

Companies

#	A	В	С	D	Е	F	G	н
T	J	К	L	м	Ν	0	Ρ	Q
R	S	T	U	V	W	х	Y	Z

Individuals/Group Contributors

#	A	В	С	D	E	F	G	Н
1	J	ĸ	L	M	Ν	0	Ρ	Q
R	S	Т	U	V	W	х	Y	Z

Products/Technologies

#	A	В	С	D	E	F	G	н
T	J	к	L	М	N	0	Р	Q

Designing

Pros:

- Built for what you need
- Manageable and extensible by you
- Minimizes Cost
- One tool for all of your platforms and teams

Cons:

- Starting Complexity
- Investment requirements
- Technical knowledge
 requirement

CourseKey's CI/CD journey

A

Alpha phase





Alpha phase

- Simple manual process
- Used FileZilla
- Forgetting to upload certain assets
- Uploading things that should not be uploaded
- Required everyone to have access to production





Beta phase

- Use build scripts!!
- Manually add files to zip folder
- Still a manual process using FileZilla



Current Phase





Phase 1

- Automated the manual process
- Develop deployment script for the server
- Look into buying a CI system



Phase 2

- Requirements
 - Bitbucket integration
 - Can run NodeJS
- Options
 - Wercker
 - Jenkins



Phase 2 gotcha

- Limited platforms support
- Immature product at the time
- Deployment support





Research





MAR 24, 2016 TUTORIAL: BUILDING WITH JENKINS INSIDE AN EPHEMERAL DOCKER CONTAINER

https://technology.riotgames.co m/news/jenkins-ephemeraldocker-tutorial

Why Jenkins

- Jenkins is extremely flexible and extensible (thousands of plugins)
- Can do Continuous integration
- Can do Continuous deployment
- Can do Reporting
- Integrates with all of our development tools and platforms

Phase 3

- Use Docker to run Jenkins
 container
- Mount Jenkins home from host machine





Jenkins



Reports





Reports

Server_Server_API_master	Ō 23.583s 🔳 4	42 Ê 1064 🌘	o 1051 🖉 Ø	 ✓ Servers / ser 100 Branch: master 	() 2m 57s	Changes Tests Artifacts No changes	S 💠 🔁 Logout
				Commit:	🕓 a day ago	Started by user Fadee Kannah	
uthController				Start	Generate Release Notes Install	Lint Tests	Deploy End
ver/api/v2/auth/auth.spec.js				•		Test winston-	
authenticate (login)				1	\bigcirc	mongodb integration	J
server/api/v2/auth/auth.spec.js						Test/Cover	×
🖲 Oms 🖹 5 🔢 5							
	121			2019			
should throw unauthorized access error if email is not register	red			0m Deploy - 1m 16s			Restart Deploy
n should throw unauthorized access error if email is not invalid	password			Om > BE - Use a	tool from a predefined Tool Insta	allation	
				> Fetches the e	nvironment variables for a giver	i tool in a list of 'FOO=bar' strings suitable	for the withEnv step.
should throw unauthorized access error if account is suspend	led			Om V > git tagcont	ains — Shell Script		
n should login the user successfully and return the right object				0m	productionci — Shell Script		1
				Send Slack M	essage		
					11 P (22)/12		
should create the correct session in the sessions collection				0m 🗸 > Publish test a	nd coverage reports — Print M	essage	
should create the correct session in the sessions collection				Om Publish test a Publish HTM	nd coverage reports – Print M L reports	essage	
should create the correct session in the sessions collection Signup()				0m × > Publish test a	nd coverage reports — Print M. L reports L reports	essage	
should create the correct session in the sessions collection signup()				Image: Omega and the set of	nd coverage reports – Print M L reports L reports kspace – Print Message	essage	
should create the correct session in the sessions collection signup()				0m Publish test a Publish test a Publish HTM Publish	nd coverage reports — Print M L reports L reports kspace — Print Message estDBs — Shell Script	essage	
should create the correct session in the sessions collection signup()				Om > Publish test a > > Publish HTM > > Publish HTM > > Publish HTM > > Clean up work > > node ./dropTit > > Recursively of	nd coverage reports — Print M L reports L reports kspace — Print Message estDBs — Shell Script elete the current directory from	essage the workspace	

Phase 3 Incidents

- Harddrive failure
- Jenkins update



Research



Phase 4

- Use Docker compose to run
 Jenkins
- Version control Jenkins home from host machine
- Version control docker configuration



Phase 4

Host machine



What is the right approach for you?

The One that works for you and your Team

Phase 4

How to use

To start

docker-compose up

To rebuild (after making changes to the dockerfiles)

docker-compose up --build

To stop the servers

docker-compose down

Phase 4

Updating

Jenkins itself

- docker pull jenkins/jenkins:lts
- docker-compose down
- docker-compose up --build

Jenkins plugins

- · Download and install through jenkins itself
- Use the following command to get the versions of installed plugins: curl -sSL "h $([^{]+})((\langle / w+ \rangle) + / 1 2 n/g'] sed 's/ /:/' > temp.txt$
 - make sure to replace username:password@localhost with the correct info
- Copy temp.txt content to online strings sort tool
- Copy sorted output to jenkins/plugins.txt
- Commit changes for jenkins/plugins.txt



Docker compose

V	ersion: '3.3'
5	ervices:
	jenkins:
	build:
	context: jenkins
	ports:
	- 80:8080
	- 32769:50000
	volumes:
	/jenkins home:/var/jenkins home
	environment:
	- TEST MYSQL=mysql
	- MONGODB HOST TEST=mongo:27017
	- MSSOL HOST=mssgl
	- FTP HOST=ftp
	depends on:
	- mysql
	- mssql
	- mongo
	- ftp
	mysql:
	image: mysql:5.7
	command:max_connections=10000sql-mode=NO_ENGINE_SUBSTITUTION
	ports:
	- "3306:3306"
	environment:
	- MYSQL_ROOT_HOST=%
	- MYSQL_ROOT_PASSWORD=
	- MYSQL_DATABASE=
	mssql:
	<pre>image: mcr.microsoft.com/mssql/server:2017-latest-ubuntu</pre>
	ports:
	- "1433:1433"
	- "1434:1434"
	restart: always
	environment:
	- ACCEPT_EULA=y
	- SA_PASSWORD=
	mongo:
	image: mongo:latest
	ports:
	- "27017:27017"

-

Jenkins Docker configuration

FROM jenkins/jenkins:lts

Update Content security policy to allow web tech execution in jenkins ENV JAVA_OPTS "-Dhudson.model.DirectoryBrowserSupport.CSP=\"default-src 'none'; ENV CI true

----- Backend Configuration -----
Set environment variables

ENV PYTHONUSERBASE /opt/python2.7/dist-packages ENV PATH="\${PATH}:\${PYTHONUSERBASE}/bin"

USER root

Install python pip for ebcli
RUN apt update && apt-get install -y python-pip
Install ebcli
RUN pip install awsebcli --upgrade --user
USER jenkins

.gitignore

ignore everything except configuration files in the root jenkins_home !jenkins_home/ jenkins_home/* !jenkins_home/*.xml jenkins_home/queue.xml

execlude critical secrets
!jenkins_home/secrets/
jenkins_home/secrets/*
!jenkins_home/secrets/master.key
!jenkins_home/secrets/hudson.util.Secret
!jenkins_home/secrets/jenkins.slaves.JnlpSlaveAgentProtocol.secret
!jenkins_home/secrets/slave-to-master-security-kill-switch
!jenkins_home/gauth/

execulde nodes configuration !jenkins_home/nodes/ jenkins_home/nodes/*.* !jenkins_home/nodes/*/config.xml

execulde users configuration !jenkins_home/users/ jenkins_home/users/*.* !jenkins_home/users/*/config.xml

execulde jobs/pipelines configurations
!jenkins_home/jobs/
jenkins_home/jobs/users

execulde ssh keys (used to connect to slaves)
!.ssh

*.idea
.DS_Store
.log

#ignore pipeline folders
branches
indexing

#ignore builds files builds lastStable lastSuccessful nextBuildNumber temp.txt

THANK YOU!

Please take a minute to rate this session in the Whova app.

- CI Workflow: https://cloud.google.com/solutions/continuous-integration/
- CommitStrip: <u>http://www.commitstrip.com/en/</u>
- Xkcd: <u>https://xkcd.com/</u>
- Awesome-CI: https://github.com/ligurio/awesome-ci
- Engineering blogs: https://github.com/kilimchoi/engineering-blogs

Fadee Kannah https://mrkannah.com





@SDStartupWeek

SAN DIEGO STARTUP WEEK